NOTARIZACIÓN DE CONSENTIMIENTOS
NOTARIZATION OF CONSENTS

TRABAJO FIN DE MÁSTER
CURSO 2019-2020

AUTOR
ALFONSO SERRANO BERMEJO

DIRECTOR
ADRIAN RIESCO RODRIGUEZ

MÁSTER EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

NOTARIZACIÓN DE CONSENTIMIENTOS
NOTARIZATION OF CONSENTS

TRABAJO DE FIN DE MÁSTER EN INGENIERÍA INFORMÁTICA
DEPARTAMENTO DE SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

AUTOR
ALFONSO SERRANO BERMEJO

DIRECTOR
ADRIAN RIESCO RODRIGUEZ

CONVOCATORIA: SEPTIEMBRE 2020
CALIFICACIÓN: 7

MÁSTER EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

23 DE SEPTIEMBRE DE 2020

RESUMEN

Notarización de Consentimientos

Con la entrada en vigor del Reglamento General de Protección de Datos (RGPD), se establece la obligatoriedad a todas las organizaciones de disponer de una autorización expresa e implícita de los usuarios de los cuales recaben y almacenen sus datos personales.

Por otro lado, para dar fe inequívoca de que esos datos han sido realmente autorizados por parte de los usuarios se propone la utilización de la tecnología Blockchain. Una de las características fundamentales de esta tecnología es la inmutabilidad del dato, siendo este en nuestro caso la propia autorización de cesión de datos personales.

En este trabajo se analizan los principales puntos de aplicación con la entrada en vigor de este Reglamento, especialmente aplicado a los sistemas web, así como un estudio de la tecnología Blockchain y de otros componentes y herramientas software tecnológicamente relacionadas con esta tecnología. Además, se realiza la implementación de una solución web que pone en práctica, de forma combinada, varias de las tecnologías estudiadas. Con esto se pretende conseguir tener una trazabilidad inequívoca de los consentimientos de cesiones de datos que pueden realizar los usuarios durante el transcurso de una navegación web.

Palabras clave: RGPD, protección datos, Blockchain, Ethereum, notarización, contrato inteligente, verificación, consentimiento.

ABSTRACT

Notarization of Consents

With the entry into force of the General Data Protection Regulation (RGPD), it is mandatory for all companies to have an express and implicit authorization from the users from whom they collect and store their personal data.

On the other hand, to unequivocally attest that these data have been authorized by users, the use of Blockchain technology is proposed. One of the fundamental characteristics of this technology is the immutability of the data, this being in our case the authorization for the transfer of personal data.

This work analyzes the main points of application with the entry into force of this Regulation, especially applied to web systems, as well as a study of Blockchain technology and other components and software tools technologically related to this technology. In addition, the implementation of a web solution is carried out that puts into practice several technologies studied in combination. This is to achieve an unequivocal traceability of the personal data transfer consents that users can make during the course of a web browsing.

Keywords: RGPD, data protection, Blockchain, Ethereum, notary, Smart Contract, notarization, consent.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

Capítulo 1 - Introducción

En la actualidad el valor de los datos personales es considerado uno de los principales valores de los que pueden disponer las empresas y organizaciones públicas y privadas. Tras la entrada en vigor del Reglamento General de Protección de Datos (RGPD) [1], estos datos personales solo pueden ser recopilados y almacenados si existe una autorización expresa por parte de las personas a las que se refieren dichos datos personales.

Por otro lado, Blockchain es una tecnología que se basa, tal y como indica su nombre, en una cadena de bloques donde se almacenan de forma codificada información sobre transacciones realizadas en la red.

Entre sus principales propiedades se encuentra la inmutabilidad del dato, por ello, una vez que se almacena la información en los nodos que forman parte de la red Blockchain, esta no se podrá ni modificar ni eliminar, solo se permitirá seguir añadiendo información. De esta forma siempre se tendrá una trazabilidad completa de la información, que permitirá asegurar la fiabilidad de las operaciones registradas.

En la etapa inicial de desarrollo de esta tecnología, Blockchain tenía su foco principal en entornos financieros y en criptomonedas, pero actualmente se ha expandido su uso a otros ámbitos como el seguimiento de transporte de mercancías (e.g. Maersk Tradelens [2]), trazabilidad de productos alimenticios (e.g. soluciones de IBM [3]), procesos electorales (e.g. piloto elecciones municipales en Denver [4]), etc.

Con este Trabajo Fin de Máster se pretende realizar un estudio de las necesidades derivadas de la aplicación del RGPD en el entorno web, y con ello plantear una solución tecnológica para que los usuarios y organizaciones empresariales puedan tener una forma fiable de verificación de aquellas autorizaciones de cesión de datos que realizan o registran.

1.1 Motivación

En este trabajo se realiza un análisis de lo siguientes puntos:

- Principales aspectos para considerar tras la entrada en vigor del RGPD.

- Estudio de la tecnología Blockchain, componentes, tipos de redes y principales plataformas disponibles.

A partir de este estudio se realiza la puesta en práctica de los dos puntos anteriores mediante implementación de un prototipo web. Para ello se han utilizado tecnologías y componentes de reciente implementación y que además están soportadas por una amplia comunidad de usuarios, que aseguran una creciente y constante evolución tecnológica.

1.2 Objetivos

El objetivo de este Trabajo Fin de Máster consiste en proponer un sistema para realizar un seguimiento de las autorizaciones de consentimientos presentes en la mayor parte de aplicaciones web en la actualidad.

A partir de lo anterior, podemos considerar los siguientes objetivos concretos:

- Identificar los requisitos derivados de la aplicación de la RGPD en el entorno web.
- Analizar principales sistemas Blockchain. Ventajas e inconvenientes de cada uno de ellos.
- Analizar principales componentes posibles para la arquitectura de solución propuesta.
- Implementación de prototipo.

1.3 Plan de trabajo

Para la consecución de los objetivos propuestos, se ha desarrollado el trabajo siguiendo el siguiente plan:

- Revisión de la aplicación del RGPD en las aplicaciones de entornos web.
- Análisis de tecnologías existentes para dar solución al objetivo buscado en este trabajo.
- Estudio de tecnología Blockchain.
- Elección de herramientas y tecnologías para implementación del prototipo.
- Desarrollo y despliegue de prototipo.
- Obtención de conclusiones.

1.4 Organización de la memoria

El presente documento se encuentra organizado en los siguientes capítulos:

- En el **capítulo 1** se realiza una introducción del objetivo del proyecto, así como del plan de trabajo para la consecución de este.
- En el **capítulo 2** se describen los conceptos principales del Reglamento General de Protección de Datos y una breve introducción del origen de la tecnología Blockchain, realizando además una explicación técnica de esta.
- En el **capítulo 3** se profundiza en la tecnología Blockchain, analizando los diversos tipos de redes existentes en la actualidad, plataformas de implementación y herramientas disponibles.
- En el **capítulo 4** se explica la implementación realizada como parte de este Trabajo Fin de Máster, dando detalles sobre la arquitectura y componentes utilizados.
- En el **capítulo 5** se incluyen las conclusiones obtenidas tras la realización de este trabajo y se proponen varias vías futuras de desarrollo.

El código fuente realizado para el desarrollo de este prototipo queda publicado en Github en la siguiente dirección:

- Proyecto web (Angular):

https://github.com/aserranob/TFM_NotarizacionConsentimientos-FrontEnd

- Servicios web (Nodejs) y Smart Contract:

https://github.com/aserranob/TFM_NotarizacionConsentimientos-BackEnd

Capítulo 2 - Estado de la cuestión

En el presente capítulo se describen los aspectos generales de aplicación a partir de la entrada en vigor del Reglamento General de Protección de Datos. Seguidamente se introducen aquí los conceptos básicos de la tecnología Blockchain.

2.1 Reglamento General de Protección de Datos (RGPD)

El Reglamento General de Protección de Datos [1] comenzó su aplicación efectiva en todos los países de la Unión Europea (UE) en el mes de mayo de 2018. Previamente las autoridades habían concedido a todos los involucrados un periodo de dos años para realizar todas las adaptaciones necesarias para su cumplimiento, periodo que va desde su entrada en vigor, en mayo de 2016 hasta la fecha indicada de aplicación efectiva.

2.1.1 Conceptos básicos

Este Reglamento es de aplicación directa en todos los países miembros de la UE, si bien, aunque no es necesario crear ninguna otra legislación a nivel nacional para transponer o desarrollar esta normativa, si se permite crear leyes nacionales para precisar algunos contenidos, entre otros, a modo de ejemplo, el RGPD permite legislar la edad a partir de la cual los ciudadanos tienen que dar su consentimiento, estando fijado un rango de años de edad entre los que cada país decide cuál quiere para su aplicación dentro de su territorio.

Esta legislación a nivel español corresponde con la Ley Orgánica de Protección de Datos Personales y garantía de los derechos digitales (3/2018), siendo aprobada por las Cortes Generales en el mes de diciembre de 2018. Con la aprobación de esta ley se derogaba la anterior Ley Orgánica de Protección de Datos (LOPD), en vigor desde su aprobación en diciembre de 1999.

2.1.2 Enfoque del RGPD

Entre las características de este RGPD se encuentra la obligatoriedad para las organizaciones que almacenen información de terceros de carácter personal, de hacerlo solo cuando haya un consentimiento inequívoco y explícito de cada persona.

Para lograr que un consentimiento sea inequívoco, ha de desaparecer toda acción por defecto o por omisión referente a la cesión de datos personales, siempre ha de existir una manifestación clara del interesado. Todos los consentimientos de cesiones de datos obtenidos por las empresas previo a la entrada en vigor del RGPD dejan de tener valor cuando estos fueron obtenidos en su momento sin autorización expresa, entre otras, obtenidos por defecto u omisión.

Además, se presta especial atención a ciertas acciones relevantes, como por ejemplo la realización de transferencias internacionales o el tratamiento de datos sensibles, donde cada persona ha de conceder de forma explícita su autorización en el momento de su cesión, no siendo permitida una autorización implícita de forma global. Por ejemplo, no se considera válido un consentimiento genérico al inicio de una navegación web, y que se mantiene durante varias páginas, donde se pueda realizar en algún momento durante la navegación una transferencia internacional.

2.1.3 Derechos del ciudadano

El RGPD mantiene los derechos ARCO (Acceso, Rectificación, Cancelación y Oposición) que existían en la LOPD y además incorpora algunos nuevos derechos, como por ejemplo la posibilidad de tener una copia de todos los datos personales y documentos que disponga el responsable, y no como ocurría con la LOPD donde solo se tenía la obligación de facilitar los datos, y no de entregar una copia de estos.

Para la gestión de todos estos derechos, los responsables de los datos pueden disponer de sistemas donde los interesados tengan acceso de forma segura y remota, de manera que por ellos mismos puedan comprobar sus consentimientos y resto de información personal.

2.2 Blockchain

Para conseguir el objetivo del proyecto, en lo referente a la notarización de los datos, nos centramos en la tecnología Blockchain, que tal y como veremos a continuación, nos ofrece tecnológicamente la inmutabilidad de los datos incluidos en las transacciones realizadas en esta red.

2.2.1 Origen de Blockchain

El término Blockchain [5] tiene su origen en el año 2008, dentro del concepto para criptomonedas Bitcoin. Con este proyecto se pretendía generar un sistema de pago electrónico, que fuera seguro, transparente y con garantía de privacidad entre sus usuarios, en el que no interviniera ninguna entidad financiera.

El objetivo fue conseguido mediante la combinación de avanzadas técnicas de criptografía y de la tecnología de las redes *Peer to Peer* (P2P).

Entre los responsables de la creación de este sistema se encuentra Satoshi Nakamoto, autor de la publicación en 2008 de "Bitcoin: A Peer-to-Peer Electronic Cash System" [6], si bien, el nombre de esta persona se cree que es un pseudónimo de la persona o grupo de personas que en definitiva se encargaron de describir el sistema en la publicación mencionada y de desplegar su implementación con el software Bitcoin.

2.2.2 ¿En qué consiste Blockchain?

El modelo de Blockchain (cadena de bloques) se basa en una red nodos donde la información se dispone en los mismos de forma descentralizada, realizándose todas las transacciones de información a través de conexiones directas. Esto iría en contraposición al modelo de internet, donde todas las transacciones y el intercambio de información se realiza de forma indirecta y centralizada, generalmente mediante un servidor que actúa de intermediario entre el emisor y receptor para almacenamiento, tratamiento y distribución de datos.

En el caso de Blockchain cada uno de los nodos se comporta de forma idéntica, ninguno tiene el control sobre el otro. Tal y como se indicaba anteriormente,

se basa en el modelo P2P, donde cada nodo actúa como cliente y como servidor a la vez.

Entre los nodos existentes en la red, tienen una tarea fundamental los llamados nodos completos o nodos validadores, que se encargan de verificar las nuevas transacciones a incorporar en la red, estos también verifican los bloques según las reglas de consenso que se hubieran establecido en la red. Estos nodos validadores, además de validar, también podrán transmitir las transacciones, teniendo además una copia completa de la red, con cada bloque y transacción.

Para poder incorporar datos a una red Blockchain, cada transacción ha de pasar por un proceso de validación, en el que forman parte todos los nodos componentes de la red, debiendo ser validada la operación por la mayoría de ellos.

A continuación, tras ser validada una transacción, entran en acción los denominados "mineros de bloques", que tienen la labor de realizar la configuración de los grupos o bloques de transacciones que sean compatibles matemáticamente con los otros bloques que hubieran sido previamente validados.

Gracias a este proceso cada bloque de la red Blockchain quedará enlazado con un bloque anterior, que permite poder relacionar de forma cronológica todas las transacciones que se han realizado en la red. De esta forma se imposibilita que un dato almacenado en un bloque anteriormente validado pueda ser alterado o manipulado sin tener una trazabilidad del cambio.

Capítulo 3 - Actividad y tipología de redes Blockchain

En la tecnología Blockchain podemos encontrar diversas tipologías de redes, que son implementadas en diferentes plataformas. En este capítulo analizamos las características de estos tipos de redes y plataformas existentes, realizando un estudio previo de las características técnicas que definen a Blockchain.

3.1 Seguridad en Blockchain

Una de las características más reconocidas de las redes Blockchain es la seguridad existente en sus transacciones, incluyendo la integridad en la información o su gran nivel de seguridad en el acceso a la red para incorporar datos o para la recuperación de estos.

La seguridad en las redes Blockchain se consigue mediante las siguientes implementaciones: encriptación de datos, claves públicas y privadas y la firma digital, que presentamos a continuación.

3.1.1 Métodos de encriptación complejos

Cada transacción u operación realizada en la red es identificada con una codificación compleja, generada por funciones *hash* que aseguran que esta sea identificada de forma única y además irrepetible.

No solo existe un hash para la transacción, sino que también cada bloque dentro de la red se encuentra identificado con otro *hash* igualmente complejo. Sumando esta codificación de bloque a la codificación de la transacción hacen que la información almacenada pueda ser verificada de forma segura.

3.1.2 Claves digitales, públicas y privadas

En Blockchain, para tener control sobre las transacciones almacenadas en la red, y para poder almacenar más información se requiere del conocimiento de claves digitales, públicas y privadas.

Asemejando estos dos tipos de clave a las transacciones financieras tradicionales, la clave pública sería el equivalente al número de nuestra cuenta bancaria, que puede ser conocido por emisor y receptor, y la clave privada sería el equivalente a nuestro código secreto para acceder a realizar operaciones sobre nuestra cuenta. En el caso de Blockchain, la clave privada permite la firma de las transacciones, identificando al emisor de forma conjunta e inequívoca con su clave pública.

Por tanto, la posesión de ambas claves deberá ser tenida únicamente por el propietario de la cuenta ya que esto significa tener un dominio completo sobre la información de la misma.

Ambas claves, pública y privada tienen relación matemática entre sí, ya que por lo general la clave pública será generada a partir de una clave privada, no siendo posible ser realizado a la inversa. Es por esto por lo que se podría verificar que una clave pública es efectivamente una clave correcta de un usuario, aunque no se podría deducir su clave privada.

3.1.3 Firma digital

La firma digital en una red Blockchain tiene como objetivo garantizar la veracidad, seguridad e integridad de los datos que se registran en los nodos de la red. De esta forma se trata de que el destinatario de un mensaje tenga la certeza de que el emisor del mismo tiene una identidad reconocida, que puede ser verificada y no falsificada. Además, con este proceso se deberá conseguir que los datos a transmitir lleguen de manera intacta a su destinatario, sin ninguna manipulación durante su transferencia. En caso de que existiese una manipulación durante el envío, el destinatario debe ser capaz de poder identificar este suceso, siendo esto posible ya que también cambiaría la firma digital.

Asimismo, el proceso de firma digital debe ser capaz de imposibilitar que un emisor pueda negar el envío de transacciones en un momento anterior. Todas las transacciones, por tanto. al ser firmadas, quedarán auditadas para una posible acción posterior.

Cada transacción que se realiza en una red Blockchain requiere de dos fases referentes a la firma digital. Estas son, la fase de firma por parte del remitente y la fase de verificación de la firma por parte del destinatario.

A partir de la clave pública y privada que tiene cada usuario entra en acción este proceso de firma digital.

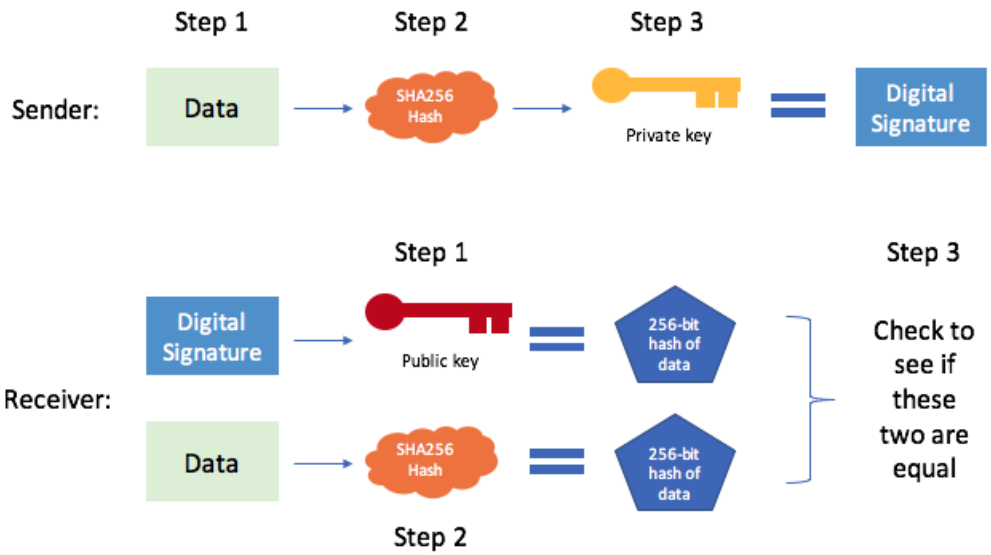

Ilustración 1. Encriptación clave pública y privada. Fuente Blair Marshall [7]

En la Ilustración 1, el emisor enviará un mensaje, y tras ser tratado con la función hash (*Step* 2), el mensaje quedará convertido en una cadena de 256 bits. A continuación (*Step* 3) esta cadena de bits será encriptada con la clave privada del emisor del mensaje. En este punto, el dato convertido en cadena de bits y encriptado ya es considerado la firma digital.

Una vez el mensaje sea recibido por el destinatario, este desencriptará la firma digital recibida con la clave pública del emisor (*Step* 1), de esta forma volverá a convertir el mensaje en el hash de 256 bits que teníamos anteriormente.

Adicionalmente, el receptor tomará los datos recibidos y realizará el mismo proceso que el emisor antes de la encriptación, es decir, el receptor realizará los pasos 1 y 2 del emisor, que consiste en aplicará la función hash hasta obtener la cadena de 256 bits.

Una vez el receptor del mensaje haga los dos procesos anteriores, comparará las dos cadenas de 256 bits, si son iguales es correcto. Si no son iguales significa que el mensaje ha sido manipulado o que el mensaje ha sido enviado por un emisor cuya clave privada con la que encriptó el mensaje no tiene relación con su pública.

Para realizar la firma digital existen varios algoritmos, siendo el más popular el *Elliptic Curve Digital Signature Algorithm* (ECDSA) [8, 9] o algoritmo de firma digital de curva elíptica.

Este algoritmo se caracteriza por la utilización de un mecanismo de encriptación llamado criptografía asimétrica, en la que la clave privada es generada a partir de la clave pública gracias a una compleja operación matemática basada en funciones de curvas elípticas. De esta forma se consigue que la firma generada a partir de la clave privada y pública sea única e irrepetible, y además, dada la complejidad computacional que supone la generación de estas claves, se imposibilita la falsificación de estas firmas digitales.

Existen otros algoritmos para firma digital, como el Schnorr [10], que agrupa las diferentes firmas digitales de los validadores de una transacción en una única firma, de forma que se reduce de manera notable el tamaño de cada transacción y de los costes derivados.

3.2 Algoritmos de consenso

Los algoritmos de consenso [11] (p.52) son métodos utilizados para apoyar toma de decisiones dentro de un grupo.

Estos algoritmos de consenso tienen una especial importancia en un entorno distribuido como las redes Blockchain públicas, descentralizadas. En estas redes los nodos tienen la obligación de ponerse de acuerdo a la hora de validar transacciones en un entorno donde los nodos no confían entre sí, y donde estos no dependen de un nodo central.

Los algoritmos por tanto tienen la finalidad de establecer unas normas, para que estas sean cumplidas y respetadas, de forma que puedan garantizar que las

transacciones registradas en la red sean realizadas de forma coherente y en condiciones de igualdad.

Para poder conseguir este consenso en este tipo de redes se hace necesario conocer con anterioridad estas reglas de consenso, es decir, se deberán conocer las reglas que tienen que cumplir los bloques para poder ser incorporados en el sistema.

Para establecer este consenso existen numerosos algoritmos, aunque en este punto nos centramos en dos de los principales, que son los de Pruebas de Trabajo y Pruebas de Participación.

3.2.1 Proof of Work (PoW) o Prueba de Trabajo.

El algoritmo de Prueba de Trabajo es el algoritmo utilizado por las principales redes públicas, entre ellas Bitcoin y Ethereum. Es el primero de los algoritmos que se crearon y es parte fundamental del proceso de minado.

En este algoritmo el proceso de minado conlleva varios intentos de creación de encriptación de claves (tentativas de *hashing*), siendo un factor importante la capacidad computacional del sistema, ya que a más capacidad mayor número de intentos por segundo. Es por este motivo por el que los mineros con una capacidad de *hash* más alta tendrán más opciones de encontrar la solución válida para el siguiente bloque, que será el denominado *hash block*.

En el algoritmo PoW los mineros tendrán la misión de validar cada bloque de transacciones para agregarlo a la red Blockchain una vez que los nodos lleguen a un consenso y acepten al denominado *hash block*.

3.2.2 Proof of Stake (PoS) o Prueba de Participación.

El algoritmo de Prueba de Participación [11] (p.52) fue implementado en el año 2011 con el objetivo de ser una alternativa al algoritmo de Prueba de Trabajo (PoW). Este algoritmo es utilizado principalmente por redes de criptomonedas, siendo la primera de ellas en utilizar este algoritmo la red Peercoin en el año 2012, posteriormente por otras como Bitshares o NXT.

Tanto este algoritmo PoS, como el anterior PoW tienen objetivos similares, aunque presentan sus principales diferencias en lo relativo a la validación de los nuevos bloques.

En este caso, en PoS, los nodos mineros son llamados nodos validadores. Aquí la decisión sobre cuál de los nodos realiza la validación de un bloque es llevada a cabo por un proceso aleatorio en el que tiene mayor peso, mayor probabilidad de hacerlo, el que cumpla unos criterios determinados. Con estos criterios se seleccionan qué nodos serán los validadores, y una vez sean elegidos, ya podrán validar las transacciones o crear nuevos bloques.

Los criterios más habituales para seleccionar los nodos validadores, aunque no únicos, son el tiempo de participación en la red o la cantidad de moneda que tiene.

Una de las consecuencias derivadas de esta característica definida en contraposición del PoW, es que en el algoritmo PoS se requiere de una cantidad de cómputo y de energía mucho menor para realizar las operaciones. Por este motivo, en la actualidad este algoritmo es cada vez más utilizado en diversos proyectos con redes Blockchain.

3.2.3 Prueba de Trabajo Programático (ProgPoW)

Este algoritmo nace como una evolución del PoW, fue diseñado para facilitar la transición de la red Ethereum desde PoW a PoS. Con esto se pretendía, tal y como se argumenta en el Git de este algoritmo [12] "*cerrar la brecha de eficiencia disponible para los ASIC especializados. Utiliza casi todas las partes del hardware básico (GPU) y viene sintonizado para el hardware más común utilizado en la red Ethereum*".

3.3 Tipos de Blockchain

Existen tres tipos de plataforma Blockchain [13], públicas, privadas y de consorcio, vemos a continuación las principales características de cada una de ellas.

3.3.1 Blockchain públicas

En este tipo de Blockchain no se requiere ningún tipo de permiso para realizar cualquier tipo de transacción en la red:

- Lectura: Cualquiera puede acceder para visualizar las transacciones que se realizan en los nodos de la red.
- Enviar transacciones: Cualquiera puede enviar transacciones a la red y esperar a su validación. En caso de ser validada, esta transacción sería añadida a la red.
- Participar en el proceso de consenso. Cualquiera puede participar en este proceso para decidir qué bloques serán añadidos en la red.

Las Blockchain públicas son redes totalmente descentralizadas, sin que exista por tanto un nodo o entidad que realice tareas de control en la red. Estas redes están basadas en el algoritmo de Prueba de Trabajo (PoW)

Este tipo de Blockchain es utilizada principalmente por las redes de criptomonedas, como Bitcoin, y por otras como Ethereum.

3.3.2 Blockchain privadas

Este tipo de red requiere de permisos para realizar operaciones de escritura en los nodos de la red. Aquí existirá por tanto un control de accesos en el que se decide quienes pueden participar en ella.

En cuanto a las operaciones de lectura, en las redes privadas se puede decidir si el acceso a los datos es también restringido o de uso público.

En estas redes además existirá una o más entidades que realizan tareas de control o validación de las transacciones. Este tipo de Blockchain es la que define redes como Hyperledger Fabric [14] o Linux Foundation [15].

3.3.3 Consorcio Blockchain o Blockchain híbrida

Una red de tipo consorcio es un híbrido entre las redes públicas y las redes privadas, combinando características de ambas. La principal diferenciación que caracteriza a estas redes se centra en el mecanismo de consenso.

Recordemos que en una red pública cualquier persona puede validar lo bloques y en una red privada esto solo lo realiza la entidad designada por la organización. Aquí, en las redes de consorcio, no se permite que cualquier persona pueda participar en el proceso que realiza la verificación de las transacciones, sino que este mecanismo de consenso es realizado por unos nodos de confianza que previamente han sido seleccionados. A diferencia de las redes privadas, estos nodos de verificación no serán de una sola organización.

En cuanto a las operaciones de lectura, al igual que en las redes privadas, puede ser configurado para su acceso de manera pública o restringida.

Este tipo de redes es considerado parcialmente descentralizado y tiene un ámbito de aplicación principalmente en el sector bancario, como ejemplo tenemos la plataforma Ethereum Alliance, donde participan el BBVA y el Banco Santander, combinando aquí la red pública Ethereum con su propia plataforma privada.

Ejemplos de este tipo de redes son BigchainDB, Evernym o r3 (basado en CORDA).

3.3.4 Blockchain como servicio (BassS)

Aunque no es un tipo de Blockchain como tal, existen empresas que ofrecen servicios de Blockchain en la red. Entre estas se encuentran IBM, Microsoft o Amazon.

En el caso de IBM o Microsoft utilizan la red Hyperledger Fabric, aunque Microsoft también tiene parte de sus servicios en redes R3 o Quorum. En cuanto a Amazon se basa en la plataforma Digital Currrency Group.

3.3.5 Comparativa entre tipos de redes Blockchain

En el siguiente cuadro podemos comprobar las diferencias entre los diferentes tipos de redes Blockchain analizados.

	Blockchain pública	Blockchain privada	Consorcio Blockchain
Requiere permisos	Si	No	No

Quién puede leer	Cualquiera	Cualquiera o Invitados	Cualquiera o Invitados
Quién puede escribir	Cualquiera	Participantes autorizados	Participantes autorizados
Rapidez en transacciones	Lento	Rápido	Rápido
Centralizada	No	Sí	Parcialmente
Quién determina consenso	Todos mineros de la red	Grupo nodos de una organización	Grupo nodos de varias organizaciones
Autorización proceso consenso	Sin autorización	Con autorización	Con autorización

Tabla 1. Diferencias entre tipos de redes Blockchain

A partir de la información anterior podemos concluir que las redes de tipología pública por sí solas no están diseñadas para un uso empresarial, ya que, tanto por rendimiento como por privacidad en los datos, no tendría un ámbito de implementación sencillo. Para el entorno empresarial por tanto sería de mayor interés una red tipo consorcio o una totalmente privada.

3.4 Principales plataformas Blockchain

Existen numerosas plataformas para implementación de redes Blockchain. A continuación, se describen algunas de las más importantes en cuanto a usabilidad y funcionalidad, y de las populares en el ámbito profesional.

3.4.1 Ethereum

Ethereum [16] es considerada en la actualidad la principal plataforma de Blockchain. Fue diseñado por Vitalik Buterin en el mes de julio de 2015 con la intención de crear aplicaciones descentralizadas más allá del uso relacionado con el pago, que hasta entonces era lo predominante en las redes existentes.

Ethereum tiene además su propia moneda llamada Ether o ETH, que tiene la utilidad de poder generar el cálculo computacional.

Es una plataforma de tipología pública, *código abierto* y descentralizada. Además, permite a los desarrolladores la creación de nuevos tipos de aplicaciones descentralizadas, llamadas DApps, que analizamos a continuación.

3.4.1.1 DApps y Contratos Inteligentes (Smart Contracts)

Ethereum permite la ejecución de contratos inteligentes (o *Smart Contracts*) en la plataforma. Un contrato inteligente es un conjunto de instrucciones ejecutables acordadas entre varias partes, y que no necesitan ser supervisadas o autorizadas por un tercero. Con la idea de estos contratos inteligentes surge una de las principales características que permite Ethereum, que es la creación de aplicaciones descentralizadas llamadas DApps.

Estas aplicaciones descentralizadas tienen las siguientes características:

- Código abierto: El código fuente debe ser accesible por todos los usuarios de la red, y estará abierto a cualquier tipo de modificación o mejora por parte de los usuarios.
- Debe tener un mecanismo de consenso incorporado. Para la validación de las transacciones puede utilizarse un protocolo basado en el algoritmo de Prueba de Trabajo (PoW) o en el de Prueba de Participación (PoS).
- Debe ser capaz de generar sus propios tokens o criptomonedas. Para validar los bloques en la Dapp son generadas dichas criptomonedas o tokens, que pueden ser propios de esa DApp o de la plataforma Ethereum (en este caso serían recompensas en Ether).
- Recompensas para mineros. Los mineros deben recibir criptomonedas o tokens al realizar sus tareas.

3.4.1.2 Herramientas de desarrollo

Para poder llegar a desplegar aplicaciones en una red Ethereum disponemos de varias herramientas que permiten a los desarrolladores el desarrollo, las pruebas y los despliegues de aplicaciones en este tipo de redes. Entre las herramientas más populares se encuentran las siguientes:

- Truffle Suite [17].

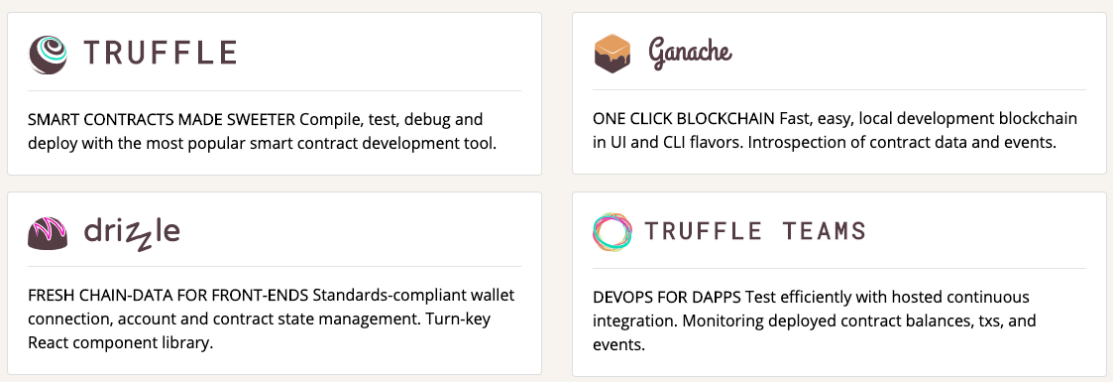

Ilustración 2. Herramientas que componen Truffle Suite [17]

Es el *framework* más popular que ofrece Ethereum para los desarrolladores que trabajan con esta plataforma Blockchain. Permite realizar lo siguiente:

- Compilar y desplegar los contratos inteligentes.
- Depurar y testear los contratos inteligentes.
- Integra facilidad para realizar despliegues y migraciones en otras redes públicas y privadas.
- Integración de contratos inteligentes mediante scripts externos.

Truffle Suite dispone de un conjunto de herramientas necesarias para poder realizar lo anterior, como son Ganache, Truffle, Truffle Teams y Drizzle.

Mediante Truffle Teams y Truffle se proporciona a cualquier DApp y contrato inteligente una serie de plantillas que permiten ser configuradas para ejecutarse y ser probadas sobre una red Ethereum, que será lanzada desde la aplicación Ganache, siendo gestionada mediante la interfaz gráfica de la herramienta Drizzle.

Ganache es por tanto una red Blockchain de Ethereum que nos podemos instalar de forma privada, con la que podemos realizar las pruebas necesarias de nuestras aplicaciones y contratos inteligentes.

Truffle Suite es una herramienta de código abierto, estando disponible para todos los interesados en GitHub [18], lo que permite que esta herramienta pueda recibir las mejoras que proporcionen los usuarios. También permite descarga herramienta en su versión estable LTS o en versiones futuras en periodo de prueba.

- Waffle [19].

  Es un framework para desarrollo y pruebas de contratos inteligentes, basado en ethers-js.

  Este framework nació con el objetivo de ser una herramienta más sencilla en su utilización, más amigable y rápida.

- Etherlime [20].

  Framework para desarrollo y despliegue de aplicaciones basado en ethers.js.

Además, disponemos de varios entornos de desarrollo (IDE), entre los que tenemos los siguientes:

- Visual Studio Code [21].

  Este entorno de desarrollo incorpora extensiones para soporte en el desarrollo de aplicaciones con el lenguaje de programación Solidity, utilizado en Ethereum.

- Remix [22].

  Este entorno de desarrollo web facilita la escritura de código Solidity para elaboración de contratos inteligentes. Permite además realizar pruebas o los despliegues de estos contratos.

3.4.2 Hyperledger

Hyperledger [14] es una plataforma de código abierto, ideada en diciembre de 2015 por la Fundación Linux y apoyada por diversas empresas del sector financiero, bancario o de telecomunicaciones.

El objetivo con el que nace Hyperledger es el poder disponer de una plataforma privada, principalmente para uso empresarial, y que fuese desarrollada

bajo estándares y protocolos abiertos en todos sus componentes (consensos, almacenamiento, autenticación, …).

Entre los principales interesados y colaboradores, que lideran esta plataforma, se encuentran empresas como IBM o Intel, aunque no es exclusivo de utilización o colaboración por parte estas compañías.

El principal motivo para disponer de una red privada, que era buscado por los interesados en la creación de esta red, se encuentra la necesidad de tener bajo control total en cada organización los datos confidenciales de los usuarios, como podrían ser los datos de transacciones bancarias o los registros de trazabilidad de ciertas operaciones empresariales.

Otro de los motivos para la creación de esta plataforma se centraba en la necesidad de disponer de una red que solventase los problemas de rendimiento y velocidad que presentaban las hasta entonces redes públicas existentes, aspecto que lograba ser mejorado a partir del diseño de Hyperledger.

Hyperledger dispone de varios *frameworks* y herramientas que facilitan su implementación y completan su funcionalidad.

Ilustración 3. Frameworks y Herramientas de Hyperledger [23]

3.4.2.1 Marco tecnológico de Hyperledger. Frameworks

Hyperledger dispone de varias tecnologías [24] en las que trabaja para completar su plataforma. Estas son las siguientes:

- Hyperledger Fabric [25].

  Es la solución más popular, permite la creación de redes privadas y de canales privados entre varias organizaciones. A partir de estos canales, de los consensos que se establezcan aquí y del lenguaje de programación elegido, se permitirá ya a continuación realizar el despliegue de los contratos inteligentes que también se contemplan en esta red, en este caso de Hyperledger, llamados Chaincodes.

- Hyperledger Burrow [26].

  Esta solución fue realizada a partir de la red Ethereum. Permite despliegues de los contratos inteligentes y el uso del lenguaje de programación Solidity.

- Hyperledger Indy [27].

  Esta solución se encarga de facilitar el desarrollo de un sistema centralizado basado en la identidad digital. Para ello ofrece una serie de librerías y componentes que son distribuidos como parte de esta plataforma.

- Hyperledger Iroha [28].

  Facilita la integración de Blockchain en ámbitos empresariales, permitiendo el despliegue de los contratos inteligentes desarrollados en otros lenguajes de programación que no son Solidity, como por ejemplo Java.

- Hyperledger Sawtooth [29].

  Desarrollado por Intel, Hyperledger Sawtooth ofrece una arquitectura modular y flexible que separa el *core* del sistema del dominio de la aplicación, de esta forma los contratos inteligentes pueden especificar las reglas de negocio para las aplicaciones sin necesidad de conocer el diseño del sistema.

  Implementa un nuevo algoritmo de consenso llamado Prueba de Tiempo Transcurrido (PoET), aunque admite otros tipos de algoritmos.

- Hyperledger Grid [30].

  Basada en estándares abiertos y buenas prácticas industriales, Hyperledger Grid ofrece una solución que proporciona los componentes necesarios para la implementación de la plataforma a nivel empresarial, como bibliotecas, modelos de datos, SDK o algunos contratos inteligentes ya basados en la lógica de negocio empresarial.

3.4.2.2 Marco tecnológico de Hyperledger. Herramientas

Las herramientas que pone a disposición de los usuarios la plataforma Hyperledger son las siguientes:

- Hyperledger Caliper [31].

  Esta utilidad nos permitirá analizar el rendimiento de cualquier plataforma Blockchain dado ciertos casos de uso.

- Hyperledger Cello [32, 31].

  Con esta solución disponemos varias herramientas para poder implementar una red Blockchain como un servicio.

- Hyperledger-Composer [33].

  Es una de las herramientas más populares, con ella podemos crear los contratos inteligentes de forma rápida y sencilla, dadas las herramientas que se disponen para facilitar este trabajo.

- Hyperledger-Explorer [34].

  Con esta utilidad podremos visualizar el estado de la red, bloques creados, estadísticas, transacciones realizadas, contratos inteligentes y demás componentes que forman parte de nuestro sistema.

- Hyperledger-Ursa [35].

  Esta herramienta nos proporciona un seguimiento de todo lo referente a la criptografía de las operaciones realizadas.

También nos permite tener una trazabilidad de todas las firmas digitales empleadas, de manera que los desarrolladores podrían realizar un análisis de estas con el objetivo de poder implementar posibles mejoras.

3.4.3 Cardano

Cardano [36] es la primera red de Blockchain de código abierto creada con un objetivo científico, siendo diseñada y desarrollada por un equipo de importantes académicos e ingenieros participantes en la comunidad Blockchain.

Esta red tiene su propia criptomoneda llamada ADA.

Cardano trabaja con un nuevo algoritmo de consenso llamado Ouroboros. Este algoritmo aplica criptografía, combinatoria y teoremas matemáticos para garantizar la longevidad y el rendimiento en las transacciones realizadas en la propia red y en las redes distribuidas que dependen de ella.

En cuanto al funcionamiento de Cardano, está desarrollado en dos capas totalmente independientes, llamadas CSL y CCL, que permite que los contratos inteligentes desarrollados sobre esta red sean más flexibles y polivalentes.

3.4.4 EOS

EOS [37] es un sistema basado en Blockchain que permite a los desarrolladores realizar el desarrollo, pruebas y ejecución de aplicaciones DApps, de forma similar a otras plataformas como Ethereum o Cardano. Además, con EOS tenemos todas las funcionalidades para crear y publicar aplicaciones, como por ejemplo, configuración de acceso a los datos, permisos, administración de la red o de las comunicaciones.

Esta red se caracteriza principalmente por lo siguiente:

- No existen comisiones para realizar transacciones, esto se consigue con los beneficios que generan los usuarios con su actividad en la red, que se convierten en beneficios a la hora de utilizar los recursos, compensando de esta forma el posible coste.

- Escalabilidad. EOS está diseñado para admitir millones de transacciones por segundo.

EOS no es considerada una red totalmente descentralizada, en este caso la red puede nombrar a algunos nodos que actúan como testigos, representando a un subconjunto de nodos en el consenso. De esta forma se trata de que no participen todos los nodos de la red en la toma de decisiones de consenso, lo que disminuye el coste y el tiempo de validación de transacciones.

Esta red trabaja con un nuevo algoritmo de consenso llamado *Delegated Proof of Stake* (DPOS). Con este algoritmo todos los usuarios de la red pueden validar sus bloques tras pasar un sistema basado en votos.

3.4.5 Corda

Corda [38] es una red de código abierto de tipología privada, los usuarios que deseen utilizarla deben estar registrados en la misma como usuarios, asegurando así la identidad de estos.

Esta red está orientada principalmente a entornos financieros. El uso principal es el de dar veracidad a contratos realizados entre dos partes, no existiendo en este caso una moneda propia, como es el caso de Ethereum, ni siquiera el concepto de criptomonedas.

Entre las características tecnológicas principales de esta red se encuentran las siguientes:

- Estabilidad. Existe un claro *roadmap* para evolución del sistema disponible en su página web. En esta evolución se asegura que con cada nueva versión se mantendrán los estándares de seguridad de consensos existentes.
- Escalabilidad. Corda está preparado para soportar gran cantidad de transacciones por segundo, del orden de miles de millones de forma diaria.

3.4.6 Resumen comparativo

A modo de resumen, en la siguiente tabla disponemos de una comparativa entre las distintas plataformas analizadas:

	Ethereum	Hyperledger	Cardano	EOS
Token	ETH	-	ADA	EOS
Organización	Ethereum Foundation	Interchain Foundation	Cardano Foundation, IOHK, Emurgo	Block.One
Objetivo	Conseguir una super red descentralizada	Crear plataforma para que las empresas puedan crear sus propias redes privadas.	Crear plataforma científica para implementación de contratos inteligentes	Crear plataforma escalable para aplicaciones distribuidas (DApps) a nivel industrial.
Consenso	POW, Casper POS.	PBFT (Practical Byzantine Fault Tolerance)	Ouroboros	DPOS
Código para contratos inteligentes	Solidity, Vyper	Lenguaje libre	Plutus	Web Assembly

Tabla 2 Comparativa entre plataformas Blockchain

Como conclusión, la determinación entre la utilización de uno u otro tipo de plataforma dependerá en primer lugar de la necesidad de disponer de una red de tipo público, privado o híbrida. A partir de aquí la elección deriva en los requerimientos que nos encontremos a la hora de desarrollar nuestra red, debiendo tener en cuenta principalmente el lenguaje de programación admitido por la plataforma y los algoritmos de consenso. Este último punto afectará a la elección de

un sistema más o menos rápido en la validación de transacciones y en tener un sistema mayor o menormente descentralizado.

Dentro de las redes públicas la que tiene un número mayor de desarrolladores activos y casos de uso contemplados es Ethereum. Para el resto de redes no públicas, entrando en aspectos técnicos, la que mayor comunidad de usuarios, herramientas y librerías dispone es Hyperledger, sin embargo la complejidad de configuración e implementación es algo mayor del resto. EOS y Cardano son similares en capacidad de velocidad, escalabilidad y rendimiento, no obstante, el algoritmo de consenso que utiliza EOS hace que se gane en algo de rapidez a costa de perder en el grado de centralización.

Capítulo 4 - Implementación

Para la puesta en práctica del objetivo definido para este Trabajo Fin de Máster se implementa un prototipo cuya arquitectura y características se exponen en este capítulo.

4.1 Arquitectura

El diagrama de arquitectura seguido para la implementación del prototipo, a alto nivel, es el siguiente:

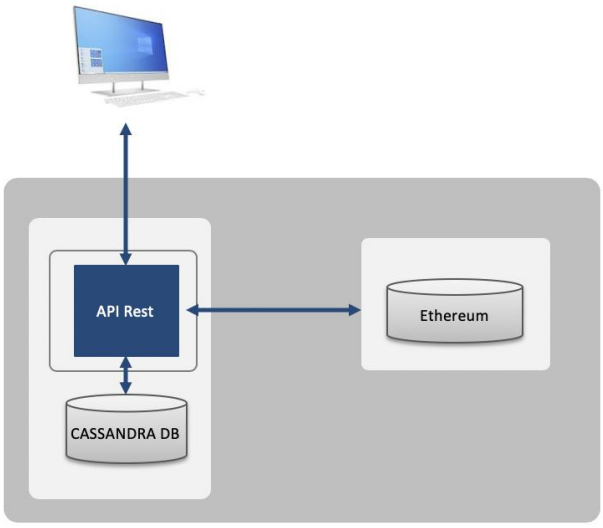

Ilustración 4. Diagrama de arquitectura prototipo

En la Ilustración 4 vemos que el sistema consta de los siguientes componentes tecnológicos:

- Aplicación web.
- Conjunto de servicios web (APIs Rest). Estas están disponibles para registro o recuperación de información en la base de datos Cassandra y en la Blockchain Ethereum.
- Base de datos Cassandra.
- Red Blockchain (Ethereum).

4.1.1 Componentes software

Los componentes utilizados para la implementación son los indicados en la Ilustración 5:

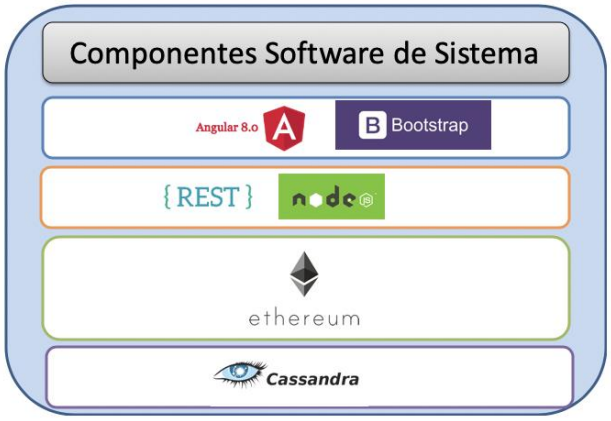

Ilustración 5. Componentes Software

Para la parte de interactuación del usuario la aplicación web está desarrollada en Angular, con componentes Bootstrap. Los servicios web utilizados están desarrollados en Nodejs, que se encargan de comunicar la aplicación web con la base de datos Cassandra y Blockchain Ethereum.

4.1.1.1 Base de datos CASSANDRA

Para almacenamiento de datos se opta por el gestor de base de datos no relacional CASSANDRA [9], especialmente diseñado para sistemas Big Data.

Los principales motivos para la elección de este gestor son los siguientes:

- Gran capacidad de escalabilidad de forma lineal y masiva. Esto nos permitirá aumentar el número de operaciones por segundo al mismo nivel que aumentemos los nodos del clúster. Es decir, si doblamos el número de nodos, doblaremos también la capacidad en número de operaciones por segundo realizadas.
- Escalabilidad de forma horizontal, permitiendo añadir nodos al clúster con hardware *commodity*, con bajo presupuesto.

- Sistema distribuido con arquitectura *Peer to Peer*. Los datos están dispuestos en todos los nodos del clúster, sin un nodo central que pueda favorecer que el sistema se vea comprometido si fallase dicho nodo.
- Alta disponibilidad. En caso de que algún nodo quede como no disponible, el sistema no se vería afectado.
- Propiedades orientado a columnas y clave – valor.
    - Un gestor de base de datos orientado a columnas almacena la información en columnas o atributos y no en filas o registros, esto nos permite tener un acceso muy rápido a la información en caso de tener grandes volúmenes de datos.
    - Una base de datos con propiedad de clave-valor dispone los datos en el sistema como un conjunto de pares clave-valor, en el que la clave servirá como identificador único de los datos almacenados. Para la obtención de la clave o del valor se puede considerar tanto un valor simple de cualquier tipo de datos hasta un objeto complejo.

En la Ilustración 6 podemos observar un ejemplo de datos almacenados como pares clave-valor en Cassandra DB.

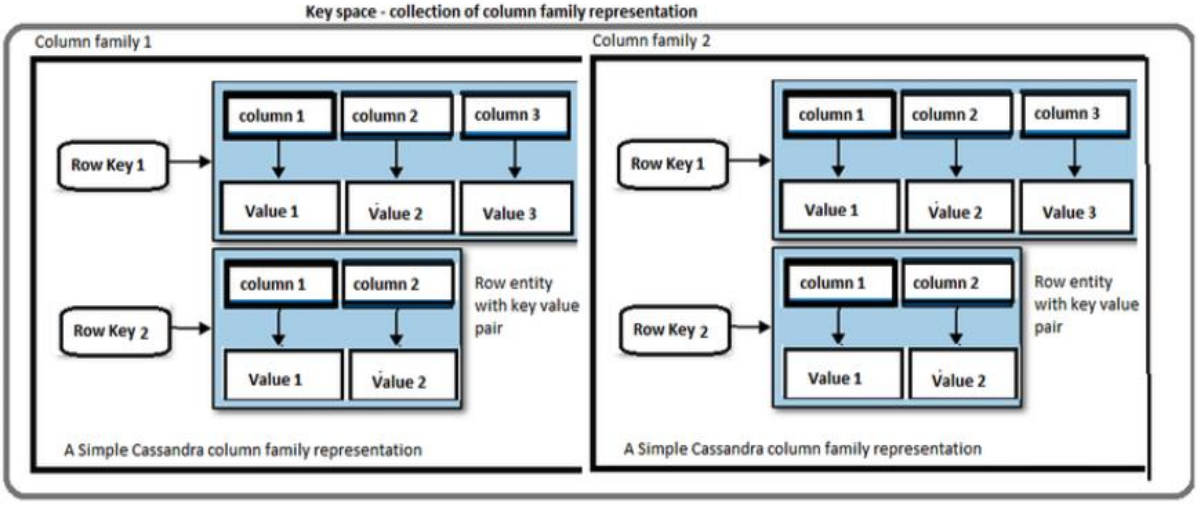

Ilustración 6. Modelo de base datos Cassandra [39]

Como consecuencia de los puntos anteriores, este tipo de base de datos tienen un rendimiento muy elevado en la extracción de datos para su utilización en aplicaciones cliente.

La finalidad de esta base de datos en el prototipo desarrollado es poder registrar los tipos de consentimientos que se van mostrando durante la navegación web, validación de usuarios y registro auxiliar de las transacciones realizadas en la red Ethereum.

4.1.1.2 Plataforma Blockchain ETHEREUM

La elección de Ethereum [16] (versión 1.9.15) como plataforma de Blockchain está basada en que, de entre todas las redes de tipología pública, esta está respaldada por una amplia comunidad de usuarios, lo que facilita la integración con otros sistemas.

4.1.1.3 Smart Contract. Remix

Realizamos un contrato inteligente para registro de las operaciones de aceptación o rechazos de consentimientos de los usuarios desde la página web.

El contrato inteligente está desarrollado con el lenguaje Solidity, con el IDE de entorno web Remix [22].

Con este contrato inteligente realizamos lo siguiente:

- Registro de información para identificar la aceptación o rechazo del consentimiento desde el *frontend*, incluyendo identificación del usuario. Los datos para conseguir este propósito son (se identifica en cursiva el nombre que corresponde en la estructura del contrato en Solidity):
    - Texto del consentimiento (*textContract*).
    - Identificador del contrato (*contractID*).
    - Identificación del usuario (*user*).
    - Aceptación o Rechazo del consentimiento (*acceptContract*).
- Obtención de datos previamente registrados en la red Ethereum.

Este contrato inteligente queda integrado en la API Rest desarrollada con nodejs, que es invocada desde la aplicación web.

4.1.1.4 Red personal Ethereum. Truffle Ganache

El *framework* Truffle Suite [17] dispone de la solución Ganache para crear redes Ethereum personales. Con esta herramienta creamos una red con varios nodos, cada uno con 100 Eth, tal y como podemos comprobar en la Ilustración 7.

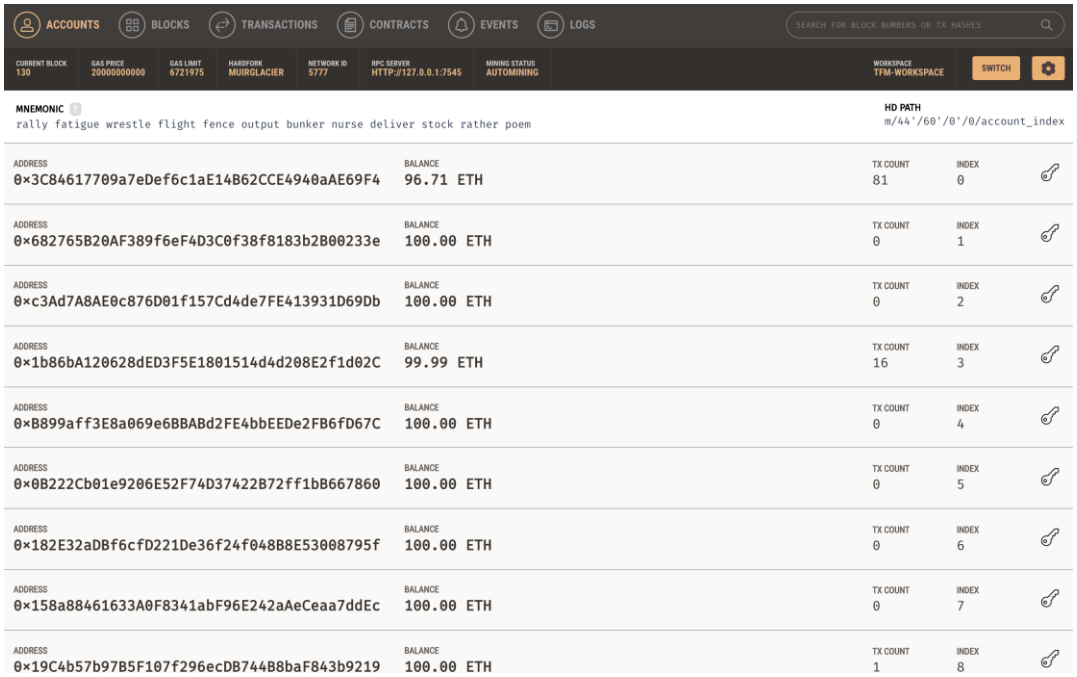

Ilustración 7. Software Ganache Ethereum

Desde esta herramienta podemos poner en práctica la visualización de todo el proceso visto anteriormente sobre esta tecnología, entre otros el registro de transacciones en Blockchain y la creación de bloques (Ilustración 8).

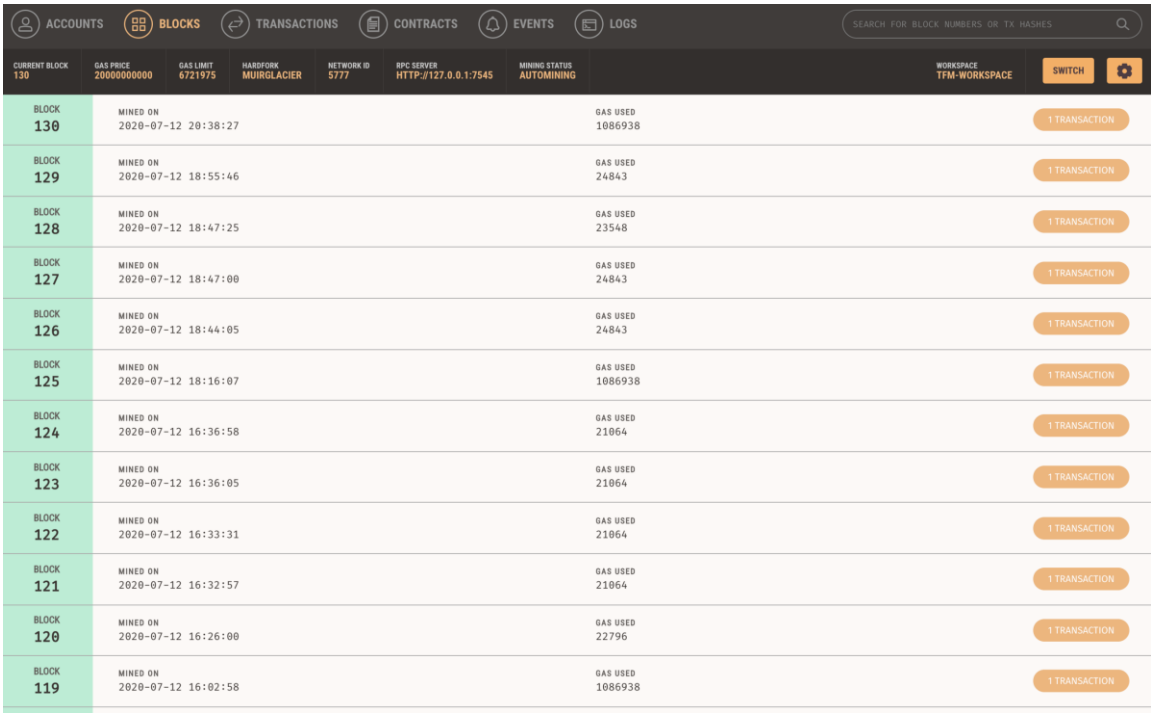

Ilustración 8. Bloques generados en red Ethereum

4.1.1.5 APIs REST. NodeJS

Para implementar la lógica de transacciones necesarias en Ethereum y en Cassandra DB se implementan varios servicios en nodejs (v. 13.2.0) [40]. Estos quedan exportados en el proyecto *BackEnd* (en la interfaz app.js), son los siguientes:

```
var contracts = express.Router();
contracts.route('/contracts').get(ContractCtrl.findAllContracts)
contracts.route('/contracts/:id').get(ContractCtrl.findAllContracts)
contracts.route('/companies').get(ContractCtrl.findAllCompanies)
contracts.route('/contractsUsers').post(ContractCtrl.contractsUsers)
contracts.route('/getSignedContracts').get(ContractCtrl.getSignedContracts)
```

Tal y como mencionábamos, estas APIs interactúan contra dos sistemas de almacenamiento, que son la base de datos Cassandra y el Blockchain Ethereum.

- CassandraDB
  - Método GET:

        - /contracts. Obtiene la información de los consentimientos que se muestran en diversos puntos de la aplicación web.
        - /companies. Obtiene la información de las compañías a las que pertenecen los consentimientos. Esto es debido a que el sistema implementado permite una gestión de consentimientos para varias compañías.
    - Ethereum
        - Método GET:
            - /getSignedContracts. Obtiene información sobre los consentimientos aceptados o denegados por los usuarios.
        - Método POST:
            - /contractsUsers. Almacena la información sobre los usuarios y consentimientos, tanto si han sido aceptados como denegados.

4.1.1.6 FrontEnd. ANGULAR

Para la implementación práctica del proceso de motorización de consentimientos, realizamos aplicación web con tecnología Angular. Desde esta aplicación el usuario puede aceptar o denegar consentimientos en diversos puntos durante su navegación por esta.

Como ejemplo, tenemos dos casos de uso presentes en prácticamente la totalidad de las páginas web actuales, que son reproducidas en la aplicación web prototipo (ver Ilustración 9):

- Aceptación de términos de uso en la participación de promociones o en registro para alta de usuario en cualquier organización.
- Aceptación de registro de cookies en el navegador mediante faldón en la parte inferior de la página web. Estas suelen registrar información del usuario durante la navegación web para personalizar publicidad, entre otras acciones.

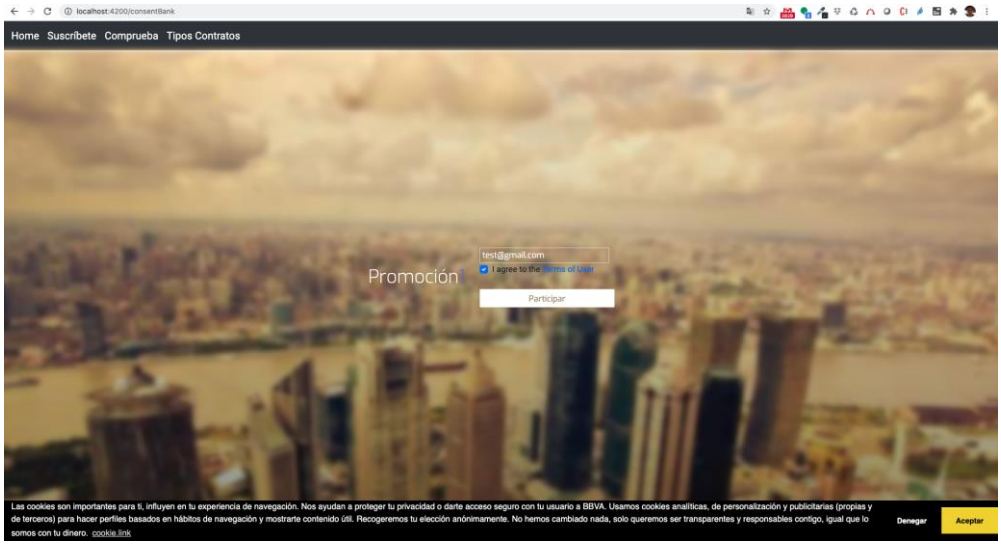

Ilustración 9. Ejemplo de consentimientos en aplicación web

Esta aplicación web tiene dependencia de las API Rest para recuperación de datos desde la base de datos Cassandra y desde la red Ethereum, por lo que, para su correcto funcionamiento, es necesario tener publicados estos *endpoints*.

4.2 Resultados

Con el desarrollo, despliegue y configuración realizada con los componentes anteriormente mencionados podemos comprobar los resultados esperados como objetivos de este trabajo, pudiendo visualizar las transacciones realizadas por los usuarios durante la navegación por la aplicación web. Estas transacciones son registradas en Ethereum, tal y como podemos visualizar en la Ilustración 10:

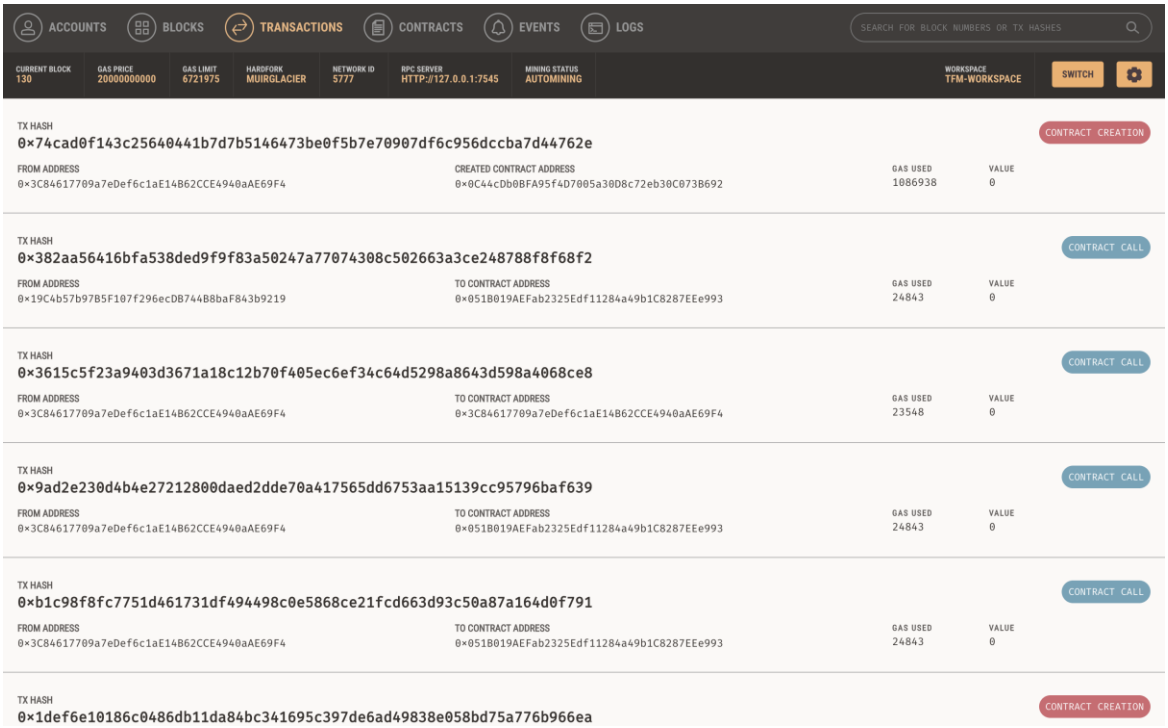

Ilustración 10. Transacciones registradas en Ethereum desde la navegación web

Capítulo 5 - Conclusiones y trabajo futuro

A continuación, se describen las conclusiones y se proponen unas líneas de trabajo futuro.

5.1 Conclusiones

Con la finalización de este Trabajo Fin de Máster se logra conseguir el objetivo que se definió inicialmente; este era poder disponer de un sistema de verificación inequívoca de consentimientos aceptados o rechazados. Para ello se ha tenido que realizar un estudio de cada una de las dos partes necesarias para finalmente poder implementar un prototipo de esta solución.

La primera de las partes que forman parte de este trabajo es el referente al Reglamento General de Protección de Datos, su ámbito de aplicación y los principales requerimientos para el tratamiento de datos personales que debe considerar cualquier organización que se disponga a solicitar a las personas o almacenar este tipo de datos en sus sistemas.

La segunda de las partes se ha centrado en el estudio de un sistema que permita dar veracidad a las operaciones relacionadas en cualquier ámbito, en nuestro caso se trataba de dar veracidad al hecho de aceptar o denegar la cesión de datos. Al fin y al cabo se trataba de buscar un sistema tecnológico que permitiese simular la acción de un notario.

Para la consecución de este objetivo se ha realizado un análisis de la tecnología Blockchain, siendo la opción más acertada para la consecución del objetivo de "*notarizar*" un hecho, ya que entre las propiedades que nos asegura esta tecnología actualmente se encuentra la inmutabilidad de las transacciones volcadas en su red.

Una vez analizadas todas las posibilidades que nos ofrece Blockchain, los tipos de redes y herramientas se opta en primer lugar por la red privada Hyperledger, y en segundo y definitivo lugar por la red de tipología pública Ethereum. El objetivo de este doble esfuerzo trataba de poder poner en práctica y completar los fundamentos teóricos desarrollados en esta memoria, incluyendo su integración con sistemas externos, tal y como se pretendía para el objetivo final.

Por último, se ha conseguido la implementación de un completo sistema donde se ha puesto en práctica el objetivo del proyecto. Para poder llegar a ello se han analizado diversas tecnologías y componentes actuales para desarrollo de aplicaciones web, para almacenamiento de datos, arquitecturas y bases de datos, además de lo anteriormente comentado sobre la tecnología Blockchain.

Sobre estos componentes tecnológicos analizados cabe remarcar la continua evolución de las versiones existentes, así como, en el caso concreto de Blockchain, de la constante aparición de nuevas plataformas. Aunque cada una de ellas surge por diferentes finalidades y diferentes grupos de usuarios, casi siempre comparten el mismo objetivo, que es el de mejorar el rendimiento y la reducción de costes, lo cual se consigue de forma casi común con la implementación de nuevos algoritmos de consenso.

De igual forma también podemos comprobar la constante aparición de nuevas herramientas que complementan y facilitan las operaciones a realizar sobre las redes Blockchain para los desarrolladores interesados en ellas. Todo ello gracias a la colaboración de la comunidad de usuarios de estas tecnologías, que principalmente permiten su evolución al ser de código abierto.

Con este análisis tecnológico se ha conseguido desarrollar e integrar todos los componentes, logrando poder disponer de una aplicación web que ofrece a las organizaciones la posibilidad de tener un sistema de verificación inequívoco de consentimientos de cesión de datos personales por parte de los usuarios de sus páginas web.

5.2 Trabajo futuro

Tomando como base este proyecto se pueden tomar varias vías de trabajo futuro.

En primer lugar, aunque este trabajo podría contemplar la verificación de consentimientos por parte de los usuarios, ha quedado más centrado en la parte de las organizaciones, por lo que sería interesante elaborar alguna funcionalidad que permitiese esta idea. El objetivo sería disponer de un sistema global de notarización de consentimientos, utilizado por tantas compañías como se suscribiesen a esta

plataforma. Los usuarios podrían ver qué consentimientos de qué empresas han sido otorgados por ellos mismos a lo largo del tiempo.

Hasta aquí la arquitectura del sistema actual y base de datos podría ser válida ya que se contempla la posibilidad de configurar N consentimientos y varias empresas, pudiendo asociar cada uno de aquellos a una empresa distinta. Pero con el trabajo futuro se pretende que para poder registrar la información se requiera de una suscripción (gratuita o no) a este sistema. Para ello será necesario al menos disponer de un sistema de securización de las APIs creadas para registro de la información, o bien convertir la red en una de tipo consorcio, híbrido entre una pública y una privada.

Otra de las posibles vías de mejora trataría sobre la posibilidad de implementar eventos que puedan realizar notificaciones a los usuarios o compañías con la autorización de consentimientos. Esta parte ha quedado en un estado inicial de desarrollo en el prototipo, considerando interesante pueda ser ampliado este Trabajo Fin de Máster con esta funcionalidad propuesta.

Capítulo 1 - Introduction

At present, the value of personal data is considered one of the main values that companies and public and private organizations can have. After the entry into force of the General Data Protection Regulation (GDPR) [1], these personal data can only be collected and stored if there is an express authorization from the people to whom said personal data refers.

On the other hand, Blockchain is a technology that is based, as its name indicates, on a chain of blocks where information about transactions carried out on the network are stored in an encrypted way.

Among its main properties is the immutability of the data, therefore, once the information is stored in the nodes that are part of the Blockchain network, it cannot be modified or deleted, it will only be allowed to continue adding information. In this way, there will always be a complete traceability of the information, which will ensure the reliability of the operations registered.

In the initial stage of development of this technology, Blockchain had its main focus on financial and cryptocurrency environments, but its use has now expanded to other areas such as tracking of freight transport (eg Maersk Tradelens [2]), traceability of food products (eg IBM solutions [3]), electoral processes (eg pilot municipal elections in Denver [4]), etc.

This Master's Thesis is intended to carry out a study of the needs derived from the application of the GDPR in the web environment, and thereby propose a technological solution so that users and business organizations can have a reliable way of verifying those authorizations of transfer of data they carry out or record.

1.1 Motivation

In this work an analysis of the following points is carried out:

- Main aspects to consider after the entry into force of the GDPR.
- Study of Blockchain technology, components, types of networks and main available platforms.

From this study, the implementation of the two previous points is carried out through the implementation of a web prototype. For this, recently implemented technologies and components have been used and which are also supported by a wide community of users, which ensure a growing and constant technological evolution.

1.2 Project objectives

The objective of this Master's Thesis is to propose a system to track consent authorizations present in most web applications today.

Based on the above, we can consider the following specific objectives:

- Identify the requirements derived from the application of the RGPD in the web environment.
- Analyze main Blockchain systems. Advantages and disadvantages of each of them.
- Analyze main possible components for the proposed solution architecture.
- Prototype implementation.

1.3 Work plan

To achieve the proposed objectives, the work has been developed according to the following plan:

- Review of the application of the GRPD in the applications of web environments.
- Analysis of existing technologies to solve the objective sought in this work.
- Blockchain technology study.
- Choice of tools and technologies to implement the prototype.
- Development and deployment of prototype.
- Obtaining conclusions.

1.4 Document organization

This document is organized into the following chapters:

- **Chapter 1** introduces the project objective, as well as the work plan to achieve it.
- **Chapter 2** describes the main concepts of the General Data Protection Regulation and a brief introduction of the origin of Blockchain technology, also making a technical explanation of it.
- **Chapter 3** delves into Blockchain technology, analyzing the various types of networks that currently exist, implementation platforms and available tools.
- **Chapter 4** explains the implementation carried out as part of this Master's Thesis, giving details about the architecture and components used.
- **Chapter 5** includes the conclusions obtained after carrying out this work and proposes several future work lines.

Source code for the development of this prototype is published on Github at the following link:

- Web project (Angular):

https://github.com/aserranob/TFM_NotarizacionConsentimientos-FrontEnd

- Web services (Nodejs) and Smart Contract:

https://github.com/aserranob/TFM_NotarizacionConsentimientos-BackEnd

Capítulo 2 - Conclusions and future work

The conclusions are described below and some lines of future work are proposed.

2.1 Conclusions

With the completion of this Master's Thesis, the objective that was initially defined is achieved; this was to be able to have an unequivocal verification system of accepted or rejected consents. For this, a study of each of the two necessary parts has had to be carried out in order to finally be able to implement a prototype of this solution.

The first part refers to the General Data Protection Regulation, its scope of application and the main requirements for the processing of personal data that any organization that is willing to request from people or store this type of data on their systems.

The second part has focused on the study of a system that allows to give truth to the related operations in any field, in our case it was to give truth to the fact of accepting or denying the transfer of data. After all, it was a question of looking for a technological system that would allow simulating the action of a notary.

To achieve this objective, an analysis of Blockchain technology has been carried out, being the most successful option for achieving the objective of "notarizing" a fact, since among the properties that this technology assures us today is the immutability of the transactions dumped on this network.

Once all the possibilities that Blockchain offers us have been analyzed, the types of networks and tools are chosen firstly for the private Hyperledger network, and secondly and definitively for the public network Ethereum. The objective of this double effort was to be able to put into practice and complete the theoretical foundations described in this report, including its integration with external systems, as was intended for the final objective.

Finally, the implementation of a complete system has been achieved where the objective of the project has been put into practice. In order to achieve this,

various current technologies and components have been analyzed for the development of web applications, for data storage, architectures and databases, in addition to what was previously commented on Blockchain technology.

Regarding these technological components analyzed, it is worth noting the continuous evolution of the existing versions, as well as, in the specific case of Blockchain, the constant appearance of new platforms. Although each of them appears for different purposes and different groups of users, they almost always share the same objective, which is to improve performance and reduce costs, which is almost commonly achieved with the implementation of new algorithms of consensus.

In the same way, we can also verify the constant appearance of new tools that complement and facilitate the operations to be carried out on Blockchain networks for developers interested in them. All this thanks to the collaboration of the community of users of these technologies, which mainly allow their evolution to be open source.

With this technological analysis, it has been possible to develop and integrate all the components, making it possible to have a web application that offers organizations the possibility of having an unequivocal verification system of consent for the transfer of personal data by users of their applications.

2.2 Future work

Based on this project, some lines of future work are suggested.

Firstly, although this work could contemplate the verification of consents by users, it has been more focused on the part of organizations, so it would be interesting to develop some functionality that allows this idea. The objective would be to have a global consent notification system, used by as many companies as they subscribe to this platform. Users could see which consents from which companies have been granted by themselves over time.

The architecture of the current system and database could be valid since the possibility of configuring N consents and several companies is contemplated, each of which can be associated with a different company. But with future work it is intended that in order to register the information a subscription (free or not) to this system is

required. For this, it will be necessary at least to have a security system for the APIs developed to record the information, or to convert the network into a consortium-type network, a hybrid between a public and a private one.

Another of the possible ways of improvement would be to implement events that can send notifications to users or companies with the authorization of consents. This part has remained in an initial state of development in the prototype, considering it interesting that this Master's Thesis can be continued with this proposed functionality.

Bibliografía

[1] «EUR-LEX.EUROPA.EU» [En línea]. Available: https://eur-lex.europa.eu/legal-content/ES/TXT/?qid=1528874672298&uri=CELEX%3A32016R0679. [Último acceso: 5 05 2020].

[2] «Maersk Tradelens» [En línea]. Available: https://www.tradelens.com. [Último acceso: 16 08 2020].

[3] «IBM Food» [En línea]. Available: https://www.ibm.com/blogs/blockchain/category/blockchain-in-food-safety/. [Último acceso: 17 08 2020].

[4] Coindesk. [En línea]. Available: https://www.coindesk.com/city-of-denver-to-pilot-blockchain-voting-app-in-coming-elections.

[5] Beginning Blockchain. A Beginner's Guide to Building Blockchain Solutions, Apress, 2018.

[6] B. A. P.-t.-P. E. C. System, «Bitcoin» [En línea]. Available: https://bitcoin.org/bitcoin.pdf. [Último acceso: 20 08 2020].

[7] «Blair Marshall» [En línea]. Available: https://medium.com/@blairlmarshall/how-does-a-bitcoin-transaction-actually-work-1c44818c3996. [Último acceso: 12 05 2020].

[8] «Academy.bit2.me.com» [En línea]. Available: https://academy.bit2me.com/que-es-ecdsa-curva-eliptica/. [Último acceso: 02 08 2020].

[9] «Apache Cassandra» [En línea]. Available: https://cassandra.apache.org.

[10] «Simple Schnorr Multi-Signatures with Applications to Bitcoin» 2018. [En línea]. Available: https://eprint.iacr.org/2018/068.pdf. [Último acceso: 20 08 2020].

[11] X. X. W. Staples, Architecture for Blockchain Applications, Springer, 2019.

[12] «Github ProgPoW» [En línea]. Available: https://github.com/ethereum/EIPs/blob/master/EIPS/eip-1057.md. [Último acceso: 01 08 2020].

[13] Blockchain-council. [En línea]. Available: https://www.blockchain-council.org/blockchain/public-vs-private-blockchain-a-comprehensive-comparison/. [Último acceso: 12 06 2020].

[14] «Hyperledger» [En línea]. Available: https://www.hyperledger.org.

[15] «Linux Foundation» [En línea]. Available: https://www.linuxfoundation.org.

[16] «Ethereum» [En línea]. Available: https://ethereum.org. [Último acceso: 01 06 2020].

[17] «Truffle Suite» [En línea]. Available: https://www.trufflesuite.com/. [Último acceso: 13 05 2020].

[18] «GitHub Truffle» [En línea]. Available: https://github.com/trufflesuite. [Último acceso: 26 04 2020].

[19] «Waffle» [En línea]. Available: https://getwaffle.io/. [Último acceso: 16 07 2020].

[20] «Etherlime» [En línea]. Available: https://etherlime.gitbook.io/etherlime/. [Último acceso: 2 07 2020].

[21] «Visual Studio Code» [En línea]. Available: https://code.visualstudio.com.

[22] «Remix Ethereum» [En línea]. Available: https://remix.ethereum.org/. [Último acceso: 12 05 2020].

[23] «Hyperledge Greenhouse» [En línea]. Available: https://www.hyperledger.org/home_aug17-2. [Último acceso: 17 08 2020].

[24] «101Blockchains» [En línea]. Available: https://101blockchains.com/es/hyperledger-blockchain-guia/. [Último acceso: 25 05 2020].

[25] «Hyperledger Fabric» [En línea]. Available: https://www.hyperledger.org/use/fabric. [Último acceso: 20 08 2020].

[26] «Hyperledger Burrow» [En línea]. Available: https://www.hyperledger.org/use/hyperledger-burrow. [Último acceso: 20 08 2020].

[27] «Hyperledger Indy» [En línea]. Available: https://www.hyperledger.org/use/hyperledger-indy. [Último acceso: 20 08 2020].

[28] «Hyperledger Iroha» [En línea]. Available: https://www.hyperledger.org/use/iroha. [Último acceso: 20 08 2020].

[29] «Hyperledge Sawtooth» [En línea]. Available: https://www.hyperledger.org/use/sawtooth. [Último acceso: 15 07 2020].

[30] «Hyperledger Grid» [En línea]. Available: https://www.hyperledger.org/use/grid. [Último acceso: 20 08 2020].

[31] «Hyperledger Caliper» [En línea]. Available: https://www.hyperledger.org/use/caliper. [Último acceso: 20 08 2020].

[32] «Hyperledger Cello» [En línea]. Available: https://www.hyperledger.org/use/cello. [Último acceso: 20 08 2020].

[33] «Hyperledger Composer» [En línea]. Available: https://hyperledger.github.io/composer/latest/. [Último acceso: 20 08 2020].

[34] «Hyperledger Explorer» [En línea]. Available: https://www.hyperledger.org/use/explorer. [Último acceso: 20 08 2020].

[35] «Hyperledger Ursa» [En línea]. Available: https://www.hyperledger.org/use/ursa. [Último acceso: 20 08 2020].

[36] «Cardano» [En línea]. Available: https://cardano.org. [Último acceso: 12 08 2020].

[37] «EOS» [En línea]. Available: https://eos.io.

[38] «Corda» [En línea]. Available: https://www.corda.net. [Último acceso: 17 05 2020].

[39] M. V, «Comparative Study of NoSQL Document, Column Store Databases and Evaluation of Cassandra» de *International Journal of Database Management Systems*, 2014, pp. 11-26.

[40] «Node.js» [En línea]. Available: https://nodejs.org/es/docs/.

[41] L. A. Bucki, Word 2013 Bible, John Wiley & Sons, 2013.

[42] CFI, «Cursos de Formación en Informática» [En línea]. Available: http://cursosinformatica.ucm.es. [Último acceso: 01 06 2019].

[43] B. Jessel, «forbes.com» *Crypto & Blockchain,* pp. Ethereum, Fabric, Corda, And Multichain. Only One Is Government Ready - New Report, 2020.

[44] C. Dannen, Introducing Ethereum and Solidity, Apress, 2017.

[45] K. Rosenbaum, Grokking Bitcoin, Manning Publications, 2018.

[46] M. Mukhopadhyay, Ethereum Smart Contract Development, Packtpub, 2018.

[47] D. Mohanty, Ethereum for Architects and Developers, Apress, 2018.

[48] A. Calder, Reglamento General de Protección de Datos (RGPD) de la UE : Una Guía de Bolsillo, IT Governance Ltd, 2017.

[49] «Agencia Española Protección Datos (AEPD)» [En línea]. Available: https://www.aepd.es/es.